

MENU

SEARCH

INDEX

DETAIL

1/1



JAPANESE PATENT OFFICE

## PATENT ABSTRACTS OF JAPAN

(11)Publication number: 09101786

(43)Date of publication of application: 15.04.1997

(51)Int.Cl.

G10H 7/02

G10H 1/00

G10H 1/00

G10H 1/02

// G10K 15/04

(21)Application number: 07258019

(71)Applicant:

MATSUSHITA ELECTRIC IND CO LTD

(22)Date of filing: 04.10.1995

(72)Inventor:

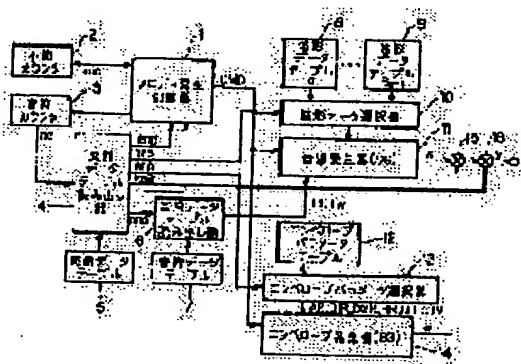
NOMURA NORIO

(54) MELODY GENERATING DEVICE BY DSP

(57)Abstract:

**PROBLEM TO BE SOLVED:** To generate a melody by the DSP with small memory capacity and a small arithmetic quantity for automatic answering telephone set, portable telephone, etc., using the DSP.

**SOLUTION:** The score of the melody is stored in a score data table 5 and a melody generation controller 1 reads a note of the score out in order; and a sound source generator(SG) 11 generates a sound source signal as a repetitive signal having the pitch cycle of the with the data on the note and an envelope generator(EG) 14 generates an envelope waveform and gives the envelope to the sound source signal, so that the DSP can generate a melody with various timbres with small memory capacity and a small arithmetic quantity.



LEGAL STATUS

[Date of request for examination] 18.09.1998  
[Date of sending the examiner's decision of rejection]  
[Kind of final disposal of application other than the  
examiner's decision of rejection or application converted  
registration]  
[Date of final disposal for application]  
[Patent number]  
[Date of registration]  
[Number of appeal against examiner's decision of rejection]  
[Date of requesting appeal against examiner's decision of  
rejection]  
[Date of extinction of right]

Copyright (C); 1998 Japanese Patent Office

MENU

SEARCH

INDEX

DETAIL



**【特許請求の範囲】**

**【請求項1】** 楽譜を音符データとして記憶する楽譜データテーブルと、前記楽譜データテーブルから読み出した音符データを順番に演奏するための制御を行う制御手段と、前記読み出された音符データが示す音符の音源信号として音符のピッチ周期の繰り返し信号を発生する音源発生手段と、前記音源信号にエンベロープを付けるためのエンベロープ発生手段と、前記音源発生手段とエンベロープ発生手段の出力を乗算してメロディ信号を出力する乗算器とを備えたDSPによるメロディ発生装置。

**【請求項2】** 乗算器から出力されたメロディ信号に音量信号を乗算する乗算器を備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項3】** 音源発生手段が音源信号を発生する際に使用する1以上の波形データテーブルを備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項4】** エンベロープ発生手段がエンベロープ波形を発生する際に使用する1以上のエンベロープパラメータテーブルを備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項5】** 各音符番号毎にピッチ周期を示す音符データテーブルを備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項6】** 楽譜データテーブルの演奏中の音符を示す音符カウンタを備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項7】** サンプリング周期毎にインクリメントされて音符長の制御に使用する小節カウンタを備えた請求項1記載のDSPによるメロディ発生装置。

**【請求項8】** 楽譜データテーブルが、音符に順番に付けられた通し番号と、音符名を表す音符番号と、音符の長さ、音符の音量と、音源発生手段が音源信号を発生する際に使用する波形データテーブルの番号と、エンベロープ発生手段がエンベロープ波形を発生する際に使用するエンベロープパラメータの番号とを備えた請求項1から7のいずれかに記載のDSPによるメロディ発生装置。

**【請求項9】** 音源発生手段が、波形データテーブルを音符の音階により決められた一定間隔毎に初めから終わりまで順次読み出し、その波形をその音符のピッチ周期毎にずらして加算することによって、その音階のピッチ周期をもつ繰り返し波形を発生する請求項1から8のいずれかに記載のDSPによるメロディ発生装置。

**【請求項10】** エンベロープ発生手段が、エンベロープ波形の立ち上がり時間と立ち下がり時間を決めるパラメータを使用してエンベロープ波形を発生する請求項1から9のいずれかに記載のDSPによるメロディ発生装置。

**【発明の詳細な説明】****【0001】**

**【産業上の利用分野】** 本発明は、DSPを使用した留守番電話、携帯電話等において、保留音等のメロディを発生するためのDSPによるメロディ発生装置に関するものである。

**【0002】**

**【従来の技術】** 近年、留守番電話等にDSPが搭載され、従来アナログ信号処理で行なわれていた機能が、DSPによるデジタル信号処理で行なわれるようになってきた。そこで、保留音等のメロディもまた、SPを使用して発生させる装置が開発されている。

**【0003】** 以下、従来のDSPによるメロディ発生装置について説明する。図30は従来のDSPによるメロディ発生装置の全体構成を示すものである。図30において、101は発振器であり、DSP動作のためのクロック信号をDSP102に供給する。102は高速演算装置であるDSPであり、ROM103に記憶された命令に従いデータの演算処理を行なう。103はROMであり、DSP102の命令およびメロディを符号化器で符号化した符号化データ等を記憶している。104はRAMであり、演算処理の途中結果等を記憶する。105は発振器であり、サンプリング周波数 $f_s$ を発生する。106はDA変換器であり、サンプリング周波数 $f_s$ の周期毎に入力のデジタル数値をアナログ信号に変換する。107はLPFであり、 $f_s/2$ 以上の周波数成分の信号を減衰させる。

**【0004】** 図31は従来のメロディ発振装置におけるDSP102の演算処理により実現されるメロディ発生のための構成である。図31(a)において、108は入力メロディ信号の符号化を行なう符号化器であり、109は符号化器108が作成した符号化データであり、ROM103に書き込んでおく。例えば、ここで符号化方式として32kbp/sのADPCM方式を使用した場合、1秒のメロディは32kbitの符号化データになる。図31(b)において、110はROM103に記憶された符号化データであり、111はROM103から読み出された符号化データを復号化するDSPの命令により実行される復号化器である。

**【0005】** 以上のように構成されたメロディ発振装置について、以下そのメロディ発生動作について説明する。メロディ発生のためには、事前に発生したいメロディを符号化器108で符号化し、その符号化データ109を作成し、ROMに書き込んでおく。メロディ演奏時には、復号化器111が、ROM103に記憶された復号化データ110を順番に読み出し、復号化を行ない、PCMのデジタル信号を出力し、DA変換器106に渡す。これにより、メロディが発生する。

**【0006】**

**【発明が解決しようとする課題】** しかしながら、上記従来のDSPによるメロディ発生装置では、メロディを音声符号化機等で符号化した符号化データをメモリに記憶

しておき、メロディ演奏時はその符号化データを復号化するので、符号化データ記憶のための多くのメモリ容量と復号化のための多くの演算量を必要とするという問題を有していた。

【0007】本発明は、上記従来の問題を解決するもので、少ないメモリ容量と少ない演算量で、多彩な音色のメロディをDSPにより発生することのできる優れたメロディ発生装置を提供することを目的とする。

【0008】

【課題を解決するための手段】本発明は、上記目的を達成するために、楽譜を音符データとして記憶する楽譜データテーブルと、楽譜データテーブルから読み出した音符データを順番に演奏するための制御を行う制御手段と、読み出された音符データが示す音符の音源信号として音符のピッチ周期の繰り返し信号を発生する音源発生手段と、音源信号にエンベロープを付けるためのエンベロープ発生手段と、音源発生手段とエンベロープ発生手段の出力を乗算してメロディ信号を出力する乗算器とを備えたものである。

【0009】

【作用】本発明は、上記構成により、楽譜データに従って各音符の信号を発生することにより、メロディの楽譜データをメモリに記憶させておけばよいので、少ないメモリ容量でメロディを発生することができる。また、従来の符号、復号化方式を使わないため、少ない演算量でメロディを発生することができる。

【0010】

【実施例】以下、本発明の実施例について、図面を参照しながら説明する。本発明におけるメロディ発生装置全体のハードウェア構成は、図22に示した従来のハードウェア構成と同様であるが、本発明では、ROM103にメロディ発生を実行するための命令および楽譜データテーブル等のデータテーブルを記憶させることと、DSP102における演算のためのアルゴリズムが従来とは異なる。

【0011】（実施例1）図1は本発明の第1の実施例におけるDSPによるメロディ発生装置の構成を示すも

のである。図1において、1はメロディ発生制御器であり、楽譜データテーブル5に記憶されている音符を順次演奏する制御を行なう。2は小節カウンタであり、その値mcはサンプリング周期毎にインクリメントされ、音符長に使用される。3は音符カウンタであり、その値ncは楽譜データテーブル5の演奏中の音符を示す。4は楽譜データテーブル読み出し器であり、音符カウンタ3の値ncが示す音符の音符データnno、leng、gain、sno、enoを楽譜データテーブル5から読み出す。楽譜データテーブル5については後述する。6は音符データテーブル読み出し器であり、音符No. がnnoの音符を音符データテーブル7から捜し、その音符のパラメータであるTp、Twを音符データテーブル7から読み出す。8、9は波形データテーブルであり、音源発生に使用する各種波形データを格納している。10は波形データ選択器であり、音源発生器11が音源発生に使用する波形データを、波形データテーブル8、9から波形データNo. snoにより選択する。11は音源発生器（SG）であり、波形データテーブル8、9のデータを使い、Tp、Twで示されるピッチ周期の繰り返し信号xを発生する。12はエンベロープパラメータテーブルであり、各種のエンベロープパラメータを格納している。13はエンベロープパターン選択器であり、エンベロープ発生器14がエンベロープ波形を発生する際に使用するエンベロープパラメータAR、D1R、D2R、RR、D1L、L1Mをエンベロープパラメータテーブル12から選択して出力する。14はエンベロープ発生器（EG）であり、エンベロープパラメータAR、D1R、D2R、RR、D1L、L1Mによりエンベロープ波形eを発生する。15、16は乗算器であり、x、e、gainの乗算を行ないメロディ信号yを出力する。

【0012】図2は楽譜データテーブル5の形式を示す。楽譜データテーブル5は、発生するメロディの楽譜を記憶しており、1行のデータが1つの音符を表し、上から順番にその音符が演奏される。データテーブルの各項目は以下のようなものである。

| 通し番号          | 音符に順番につけられた番号                                 |
|---------------|---|
| 音符名           | 音符の名前   |
| 音符No. nno     | 音符名に付けられた音符を表す音符番号                            |
| 長さ leng       | 音符の長さ   |
| ゲイン gain      | 音符の音量   |
| 波形データNo. sno  | 音源発生器11が音源信号を発生する際に使用する波形データテーブルの番号           |
| エンベロープNo. eno | エンベロープ発生器13がエンベロープ波形を発生する際に使用するエンベロープパラメータの番号 |

波形データNo. snoとエンベロープNo. enoにより、その音符の音色が決定される。

【0013】図3は図4に示す楽譜のための楽譜データ

テーブル5の具体例を示している。例えば、通し番号=1、音符名=G5、音符No.=32、長さ=24は、5オクターブ目のG音で二分音符を示し、通し番号=3

1、音符名＝\*\*、音符No.＝38、長さ＝12は、四分休符を示し、通し番号＝63、音符名＝ED、音符No.＝39、長さ＝0は、曲の終わりを示す終止符を示す。楽譜データテーブル5の最終行は、音符名＝ED、音符No.＝39でなくてはならない。またここでは、長さlengの値は、1小節の長さを48としているので、全音符が48、2分音符が24、四分音符が12になる。なお、ここでは楽譜データテーブル5に音符名を記憶しているが、これは説明用であり、音符No.だけで足りるので、実際には必要はない。

【0014】図5は音符データテーブル7の形式を示す。1行が1つの音符を表し、音符No. 毎に、音符名、Tp、Twが記憶されている。Tp、Twの値はその音符のピッチ情報を表し、音源発生器11が音源信号を発生するときのパラメータとして使用される。

【0015】図6は音符データテーブル7の具体例を示す。例えば、音符No. 32は、音符名がG5で、ピッチ情報Tpが653、Twが1606を示し、音符No.＝38、音符名＝\*\*は四分休符を示し、音符No.＝39、音符名＝EDは曲の終わりを示す。なお、ここでは音符データテーブル7に音符名を記憶しているが、これは説明用であり、実際には必要はない。

【0016】図7は波形データテーブル8、9を示す。この波形データは、音源発生器11が音源信号xとして繰り返し波形を発生するとき使用される。図7では、

$$\begin{aligned} mc &= 0 \\ mc &= leng * Temp - 1 \\ mc < mc < leng * Temp - 1 \end{aligned}$$

ここで、Tempは曲のテンポを示す定数である。四部音符の長さをTとすると、

$$Temp = (fs \times 4) / T \times L$$

である。ここで、

fs サンプリング周波数 [Hz]

L 1小節の長さを表すleng (図2の楽譜データテーブルではL=48)

【0019】ST3. 1)  $mc = 0$ の時  
新しい音符の信号を発生する。

ST3. 1. 1) 楽譜データを読み出す。

楽譜データテーブル読み出し器4が、楽譜データテーブル5から音符カウンタ3の値ncで示される以下の音符データを読み出す。

nno 音符No.

leng 長さ

gein ゲイン

sno 波形データNo.

eno エンベロープNo.

ST3. 1. 2) 音符データテーブルを読み出す。

音符データテーブル読み出し器6が、音符データテーブル7から音符Noがnnoである音符の以下のパラメータを読み出す。

波形データテーブルはSn個存在している。

【0017】図8はエンベロープパラメータテーブル12の例を示す。このテーブルは、1行で1つのエンベロープパラメータAR、D1R、D2R、RR、D1L、L1Mを記憶している。これらのパラメータは、エンベロープ発生器13がエンベロープ波形eを発生するときのパラメータとして使用される。図8では、エンベロープパラメータ数は3組である。

【0018】図9は本実施例におけるメロディ発生方法のためのメロディ発生制御器1におけるアルゴリズムを示す。以下、このアルゴリズムの動作を図1のブロック図に対応させて説明する。

ST1) 初期化を行なう。

以下のように初期化を行なう。ここで、終了フラグは、楽譜データテーブル5の最後まで演奏が終わった時に1になるフラグである。

小節カウンタmc=0、音符カウンタnc=1、終了フラグ=0

ST2) 終了フラグを調べる。

終了フラグ≠0のときST8へ分岐する。

ST3) 音源発生器(SG)13とエンベロープ発生器(EG)14に動作コマンドCMDを設定する。

mcの値により、SG、EGに与える動作コマンドCMDおよびパラメータを設定する。まずmcの値を調べる。

の時ST3. 1へ分岐する。

の時ST3. 2へ分岐する。

の時ST3. 3へ分岐する。

Tp、Tw 音符のピッチ情報

ST3. 1. 3) 波形データテーブルを選択する。

波形データ選択器10が、波形データNo. snoにより、音源発生器11が使用する波形データテーブルを選択する。

ST3. 1. 4) エンベロープパラメータテーブルを読み出す。

エンベロープパラメータ選択器13が、エンベロープパラメータテーブル12からエンベロープNo. enoの以下のパラメータを選択し出力する。

AR、D1R、D2R、RR、D1L、L1M エンベロープ波形を決めるパラメータ

ST3. 1. 5) 音符の種類を調べる。

音符No. nnoの値を調べ新しく演奏する音符の種類を調べ、音源発生器(SG)11、エンベロープ発生器(EG)14に動作コマンドCMDを設定する。

発生音有り ( $1 \leq nno \leq 37$ )の時 3. 1. 5. 1へ分岐する。

休符 ( $nno = 38$ )の時 3. 1. 5. 2へ分岐する。

終了 ( $nno = 39$ )の時 3. 1. 5. 3へ分岐する。

動作コマンドCMDは以下のである。

“ON” 新しい音符の信号を発生  
 “OFF” 現在の音符の信号を発生を終了  
 “HOLD” 現在の音符の信号を発生を継続

【0020】

ST3. 1. 5. 1) 発生音あり ( $1 \leq nno \leq 37$ )  
 SG11、EG14に“ON”コマンドを設定する。ST4へ分岐する。

ST3. 1. 5. 2) 休符 ( $nno == 38$ )  
 SG11、EG14に“OFF”コマンドを設定する。ST4へ分岐する。

ST3. 1. 5. 3) 終了 ( $nno == 39$ )  
 SG11、EG14に“OFF”コマンドを設定する。  
 終了フラグ=1

ST4へ分岐する。

【0021】

ST3. 2)  $mc == leng * Temp - 1$ の時  
 現在演奏している音符の最終のサンプル値を発生する。  
 SG11、EG14に“HOLD”コマンドを設定する。

小節カウンタ  $mc = 0$

音符カウンタ  $nc = nc + 1$

ST4へ分岐する。

ST3. 3)  $0 < mc < leng * Temp - 1$ の時  
 現在演奏している音符を継続して発生する。SG11、EG14に“HOLD”コマンドを設定する。

小節カウンタ  $mc = mc + 1$

ST4へ分岐する。

【0022】ST4) 音源発生器 (SG) 11が音源データを発生し  $x$  を出力する。

動作コマンドCMDとパラメータ  $Tp$ 、 $Tw$  波形データテーブルにより、サンプリングタイミング毎に繰り返し波形である音源信号  $x$  を発生する。

ST5) エンベロープ発生器 (EG) 14がエンベロープ波形を発生し  $e$  を出力する。

動作コマンドCMDとエンベロープパラメータ  $AR$ 、 $D1R$ 、 $D2R$ 、 $RR$ 、 $D1L$ 、 $L1M$ により、サンプリングタイミング毎にエンベロープ波形  $e$  を出力する。

ST6) 出力値  $y$  を計算する。

$y = x * e * gain$

ST7) 次のサンプリングタイミングまで待つ。

ST2~7の処理がサンプリング周期毎に実行されるようにする。

ST8) メロディ発生終了

【0023】図10は上記図9のアルゴリズムの動作の様子を示す。各音符の最初のサンプリングタイミングで、図1のメロディ発生制御器1は、動作コマンドCMDとして“ON”コマンドまたは、“OFF”コマンドを出力する。その他のサンプリングタイミングでは、“HOLD”コマンドを出力する。図1の音源発生器

(SG) 11は、その音符の音程に対応する繰り返し信号を音源信号  $x$  として出力する。繰り返し信号のピッチ周期は、パラメータ  $Tp$ 、 $Tw$  により決定される。図1のエンベロープ発生器 (EG) 14は、エンベロープ波形  $e$  を発生する。エンベロープ発生器14は、“ON”コマンドを受け取ると、決まった波形の出力を開始する。エンベロープ波形の形は、 $AR$ 、 $D1R$ 、 $D2R$ 、 $RR$ 、 $D1L$ 、 $L1M$  により決定される。図1の乗算器15、16は、 $x$ 、 $e$ 、 $gain$  を掛け、メロディ信号  $y$  を出力する。

【0024】このように、上記第1の実施例によれば、楽譜データテーブル5、音符データテーブル7、波形データテーブル8、9、エンベロープパラメータテーブル12をROMに記憶し、これらをメロディ発生制御器1が制御することにより、DSPによるメロディを発生することができる。また、これらのデータテーブルが使用するROM容量は、従来の装置で使用するROM容量より少ない。さらに、図9で示されるDSPの処理の演算量は、従来の音声符号、復号化方式より少ない。その他以下の特徴がある。

1) 楽譜データテーブル5を書き換えることにより、簡単にメロディの変更ができる。

2) 波形データテーブル8、9およびエンベロープパラメータテーブル12を書き換えることにより、簡単に音色の変更ができる。

【0025】なお、上記第1の実施例では、波形データテーブル8、9は複数個の波形を記憶しているが、最低1つあればよい。このとき、波形データ選択器10は必要ない。またエンベロープパラメータテーブル12はエンベロープパラメータを複数組記憶しているが、最低1組あればよい。このとき、エンベロープパラメータ選択器13は必要ない。さらに、エンベロープ発生器 (EG) 14を使用せず、 $e = 1.0$  としても、音階の発生はできる。さらに、楽譜データテーブル5に  $Tp$ 、 $Tw$  のパラメータを記憶させるようにすれば、音符データテーブル読み出し器6および音符データテーブル7は必要なくなる。

【0026】(実施例2) 次に、本発明の第2の実施例について、図面を参照しながら説明する。本実施例におけるメロディ発生装置の全体構成は図1に示した第1の実施例と同じであるが、本実施例ではその中の波形データテーブル8、9と波形データ選択器10と音源発生器11とからなる音源発生部を別の構成としたものである。まず本実施例における動作原理について説明する。

【0027】図11は繰り返し波形発生原理を示す。図11の  $h(t)$  はインパルス応答波形を表し、 $sp(t)$  はピッチ周期が  $Tp$  のインパルス列を表す。一番下の繰り返し波形  $x(t)$  は、 $h(t)$  と  $sp(t)$  の畳み込み演算により求められる。

$x(t) = h(t) * sp(t)$

ここで、\*は畳み込み演算を表す。つまり、 $h(t)$ を $T_p$ ずつずらした信号を加算することでピッチ周期が $T_p$ の繰り返し波形 $x(t)$ を得る。図11においては、 $s_1(t) \sim s_6(t)$ が $h(t)$ を $T_p$ ずつずらした信号であり、これらを加算することで繰り返し信号 $x(t)$ を得ている。

【0028】図12はピッチ周期が $T_{p'}$ の繰り返し波形発生の様子を示している。ここで、インパルス応答波形 $h'(t)$ はインパルス応答波形 $h(t)$ を時間軸方向に $T_p/T_{p'}$ でスケーリングしたものである。  

$$h'(t) = h(t \cdot T_p / T_{p'})$$

【0029】図13は図11の様子を周波数軸上で表したものである。時間領域の信号 $h(t)$ 、 $s_p(t)$ 、 $x(t)$ を周波数領域に変換したスペクトルが $H(\omega)$ 、 $SP(\omega)$ 、 $X(\omega)$ である。 $X(\omega)$ は、基本周波数が $\omega_p$ で、その倍の高周波成分を持つスペクトルとなる。

【0030】また、図14は図12の様子を周波数軸上で表したものである。図13の $x(\omega)$ と図14の $x'(\omega)$ は基本周波数は異なるが同じ倍音構成を持つ。このように、インパルス応答 $h(t)$ を上記のようにスケーリングすることにより、同じ倍音構成でピッチ周期が異なる繰り返し信号を得ることができる。これにより、同様の音色で異なるピッチ周期の信号を発生することができ、同様の音色の音階を発生することができる。

【0031】図15は本実施例における音源発生部の構成を示す。図15において、17は音源制御器、18はCH1波形データ読み出し器、19はCH2波形データ読み出し器、20はこれら2つの波形データ読み出し器の出力 $x_1$ 、 $x_2$ を加算して $x$ を出力する加算器、21は2つの波形データ読み出し器18、19が読み出す波形データを格納した図7と同じ波形データテーブルである。

【0032】音源制御器17は、動作コマンドCMDとピッチ周期 $T_p$ により2つの動作CHの制御を行ない、以下の制御変数を持つ。

$posp$       ピッチ周期の制御用の変数。サンプリング周期毎に64ずつ加算される。 $posp \geq T_p$ となると、 $posp = posp - T_p$ とし、“停止状態”のCHを捜しそのCHを“動作状態”にする。

$SG.stt$       音源の動作状態で“停止状態”、“動作状態”を示す。

【0033】2つの波形データ読み出し器18、19は、それぞれ独立に波形データテーブル21を初めから終わりまで $Tw$ 間隔で順番に読み出す。 $CH_n$  ( $n=1, 2$ )には以下の制御変数がある。

$CH_n.pos$       波形データテーブル21の読み出し位置を示す。 $Tw$ ずつ加算されることにより、波形データテーブル21をスケーリングして読み出す。波形データテーブル21の最終まで読み出すとCH動作状態SH

$n.stt$  “停止状態”にする。

$SG.stt$       CHの動作状態を表し“動作状態”、“停止状態”を表す。

【0034】図16は2つの波形データ読み出し器18、19が波形データテーブル21を読み出す様子を示し、それぞれパラメータ $Tw$ により、波形データテーブル21をスケーリングして読み出す動作を行なう。ここでは、CHを2つ使用し、それぞれの出力を、 $x_1$ 、 $x_2$ としている。

【0035】以下、本実施例における音源発生部の動作を図17～図20を用いて説明する。図17の処理は、図9のST4の[SGEN]の処理である。

【0036】ST11) 音源CHの制御を行なう。  
 図17の[SGCHCNT]の処理を図18に示しており、これは図15の音源制御器17の動作に対応している。

ST11. 1) 動作コマンドCMDを調べる。

動作コマンドCMD==“OFF”の時 ST11. 1. 1へ分岐する。

動作コマンドCMD==“ON”の時 ST11. 1. 2へ分岐する。

動作コマンドCMD==“HOLD”の時 ST11. 1. 3へ分岐する。

ST11. 1. 1) 動作コマンドCMD==“OFF”の時

CH1動作状態CH1.  $stt$  = “停止状態”

CH2動作状態CH2.  $stt$  = “停止状態”

音源状態SG.  $stt$  = “停止状態”

ST12へ分岐する。

ST11. 1. 2) 動作コマンドCMD==“ON”の時

CH1、CH2を初期化する。

CH1動作状態CH1.  $stt$  = “停止状態”

CH2動作状態CH2.  $stt$  = “停止状態”

音源状態SG.  $stt$  = “動作状態”

ST12へ分岐する。

ST11. 1. 2) 動作コマンドCMD==“HOLD”の時

音源状態SG.  $stt$ を調べる。

音源状態SG.  $stt$  = “動作状態”の時 ST11. 1. 3. 1へ分岐する。

音源状態SG.  $stt$  = “停止状態”の時 ST2へ分岐する。

ST11. 1. 2) “動作状態”の時

$posp = posp + 64$

$posp \geq T_p$ の時、以下の動作をする。

$posp = posp - T_p$

停止状態のCHを捜し、そのCHの番号を $n$ とする。  
 $CH_n.pos$ に初期値を設定し、そのCHを“動作状態”にする。



$CHn.pos = (int) (posp * Tp / 64)$   
 $CHn.stt$ を“動作状態”にする。

【0037】ST12) 音源CH1の動作を行なう。

図17の[SGH10OR]の処理を図19に示しており、これは図15のCH1波形データ読み出し器18の動作に対応している。

ST12.1) CH1の動作状態 $CH1.stt$ を調べる。

$CH1.stt = \text{“動作状態”}$ の時 ST12.1.1へ分岐する。

$CH1.stt = \text{“停止状態”}$ の時 ST12.1.2へ分岐する。

ST12.1.1)  $CH1.stt = \text{“動作状態”}$ の時  $CH1.pos$ の値を調べる。ここで、 $TL$ は波形データテーブルの大きさを表す。

$CH1.pos \geq TL$ の時 ST12.1.1.1へ分岐する。

$CH1.pos < TL$ の時 ST12.1.1.2へ分岐する。

ST12.1.1.1)  $CH1.pos \geq TL$ の時 データテーブル21の読み出しを終了する。

$x1 = 0$

$CH1$ の動作状態 $CH1.stt = \text{“停止状態”}$

ST13へ分岐する。

ST12.1.1.2)  $CH1.pos < TL$ の時 波形データテーブルの $CH1.pos$ の位置のデータを読み出し、結果を $x1$ に出力する。

$CH1.pos = CH1.pos + Tw$

ST13へ分岐する。

ST12.1.2)  $CH1.STT = \text{“停止状態”}$ の時

$x1 = 0$

ST13へ分岐する。

【0038】ST13) CH2の動作を実行する。

図17のST13の[SGH2OPR]の処理を図20に示しており、これは図15の2波形データ読み出し器19の動作に対応しており、CH1波形データ読み出し器18と同様の動作を行ない、結果を $x2$ に出力する。

【0039】ST14) CH1、CH2の出力 $x1$ 、 $x2$ を加算器20で加算する。

$x = X1 + X2$

【0040】図21は図15の2つの波形データ読み出し器18、19が周期が $Tp$ の繰り返し波形 $x(t)$ を作成する動作を示す。

【0041】図22は $Tp = 100$ 、 $Tw = 10$ のときの音源発生部の動作例を示す。ここで、 $h[i]$ は波形データテーブルを示している。CH1波形データ読み出し器18では、波形データテーブル $h[i]$ の読み出し位置の初期値は0であり、読み出し間隔は $Tw = 10$ である。つまり、 $i = 0, 10, 20, \dots$ の位置のデ

ータをサンプリングタイミングに順次 $x1$ として出力する。CH2波形データ読み出し器19は、以下の値がデータテーブル読み出しの位置の初期値である。

$(int) ((128 - Tp) / 64 * Tw) = 4$

したがって、CH2は波形データテーブル $h[i]$ の $i = 4, 14, 24, \dots$ の位置のデータをサンプリングタイミングに順次 $x2$ として出力する。

【0042】上記のような処理を行なうことにより、音源発生部の出力信号のピッチ周期 $Tp$ をサンプリング周期の $1/64$ の精度で設定できる。サンプリング周波数 $fs = 8kHz$ の場合、この程度の精度でピッチ周期 $Tp$ を設定できないと、正確な音階のピッチを発生できない。各音符の $Tp$ の値は以下のように計算される。

$Tp = fs \times 64 / f$

ここで、

$fs$  サンプリング周波数 [Hz]

$f$  音符のピッチ周波数 [Hz]

例えば、A4の音符の場合、 $f = 440$  [Hz]であり、 $fs = 8000$  [Hz]とすると $Tp = 1164$ となる。また、図6の音符データテーブルの $Tw$ の値は以下の始期で計算されている。

$Tw = f \times 16384 / fs$

ここで、16384は任意の定数である。ここでは、波形データテーブルの大きさを16384とし、ピッチ周期で波形データテーブルの読み出しが終了するように定数を決めた。ここでは、波形データテーブルの大きさを16384としたが、1次近似による内挿を行なうことで実際の波形データテーブルの大きさを256にすることができる。

【0043】以上のように、上記第2の実施例によれば、波形データテーブル21の内容は任意であり、どんな内容であってもパラメータとして与えられたピッチ周期 $Tp$ を持つ繰り返し信号を発生することができるため、波形データテーブル21の内容により、多様な倍音構成の繰り返し信号を発生することができる。これにより多様な音色を発生することができる。また、図17～図20で示されるDSP演算の処理量は少なくてもよい。

【0044】(実施例3) 次に、本発明の第3の実施例について、図面を参照しながら説明する。本実施例は、図1に示したエンベロープ発生器(EG)14の別の動作例を示すものである。

【0045】図23は本実施例におけるエンベロープ発生器(EG)の動作のアルゴリズムを示す。図23の処理は、図9のST5の[EGEN]の処理である。エンベロープ発生器(EG)は、“ON”コマンドによりエンベロープ波形の発生を開始する。波形の形状はパラメータAR、D1R、D2R、D1L、RR、LIMにより決定される。

【0046】図24(a)は“ON”コマンドにより発生されるエンベロープ波形 $e$ を示す。また、図24

(b) はエンベロープ波形発生途中で“OFF”コマンドを受けた時のエンベロープ波形 $e$ を示す。

【0047】図25はエンベロープ波形発生の状態遷移図を示す。図25の各状態では、エンベロープ発生器(EG)は以下の動作を行なう。

1) “AR状態”

エンベロープ波形 $e$ の立ち上がり部分を発生する。立ち上がりの速さはARにより決定される。 $e \geq 1.0$ となると $e = 1.0$ として、“D1R状態”に遷移する。図26に動作アルゴリズムを示す。“AR状態”では、 $e$ はサンプリングタイミング毎に以下のように変化する。

$$e = e + (2.0 - e) * AR$$

2) “D1R状態”

エンベロープ波形 $e$ の第1の立ち上がり部分を発生する。立ち上がりの速さはD1Rにより決定される。 $e \leq D1L$ となると $e = D1L$ として、“D2R状態”に遷移する。また途中で $e \leq LIM$ となると $e = 0$ として、“OFF状態”に遷移する。図27に動作アルゴリズムを示す。“D1R状態”では、 $e$ はサンプリングタイミング毎に以下のように変化する。

$$e = e * D1R$$

3) “D2R状態”

エンベロープ波形 $e$ の第2の立ち上がり部分を発生する。立ち上がりの速さはD2Rにより決定される。 $e \leq LIM$ となると $e = 0$ として、“OFF状態”に遷移する。図28に動作アルゴリズムを示す。“D2R状態”では、 $e$ はサンプリングタイミング毎に以下のように変化する。

$$e = e * D2R$$

4) “RR状態”

“AR状態” “D1R状態” “D2R状態” のいずれかの状態において、“OFF”コマンドを受け取ると“RR状態”になる。この状態はエンベロープ波形 $e$ の立ち下がり部分を発生し、立ち下がりの速さはRRにより決定される。また、 $e \leq LIM$ となると $e = 0$ として、“OFF状態”に遷移する。図29に動作アルゴリズムを示す。“RR状態”では、 $e$ はサンプリングタイミング毎に以下のように変化する。

$$e = e * RR$$

5) “OFF状態”

$e = 0$ とする。

【0048】図8のエンベロープパラメータテーブルにおいて、エンベロープNo. = 0のパラメータは持続音を発生し、エンベロープNo. = 1, 2は減衰音を発生する。

【0049】以上のように、上記第3の実施例によれば、AR、D1R、D2R、RR、D1L、LIMのパラメータにより簡単なアルゴリズムで、エンベロープ波形を発生することができる優れたエンベロープ発生器を実現できるものである。

【0050】

【発明の効果】本発明は、上記第1の実施例から明らかなように、以下に示す効果を有する。

- 1) 楽譜の形でメロディを記憶しているため、必要とするROM容量が少ない。
- 2) 従来の符号、復号化方法を使用しないため、必要とするDSPの演算量が少ない。
- 3) 楽譜データテーブルを書き換えることにより、簡単にメロディを変えることができる。
- 4) 波形データテーブル、エンベロープパラメータテーブルを書き換えることにより、簡単に音色を変えることができる。

【0051】本発明はまた、上記第2の実施例から明らかなように、以下に示す効果を有する。

- 1) 波形データテーブルの内容により、多くの音色の音を発生することができる。
- 2) サンプリング周期より細かくピッチ周期を設定することができる。
- 3) 簡単なアルゴリズムでDSPにより実現することができる。

【0052】本発明はまた、上記第3の実施例から明らかなように、以下に示す効果を有する。

- 1) 簡単なアルゴリズムでDSPによりエンベロープ波形を発生することができる。

【図面の簡単な説明】

【図1】本発明の第1実施例におけるDSPによるメロディ発生装置のブロック図

【図2】本発明の第1実施例における楽譜データテーブルを示す一覧図

【図3】本発明の第1実施例における楽譜データテーブルの具体例を示す一覧図

【図4】本発明の第1実施例における楽譜データテーブルの具体例の基になる譜面図

【図5】本発明の第1実施例における音符データテーブルの一覧図

【図6】本発明の第1実施例における音符データテーブルの具体例を示す一覧図

【図7】本発明の第1実施例における波形データテーブルの例を示す波形図

【図8】本発明の第1実施例におけるエンベロープパラメータテーブルの例を示す一覧図

【図9】本発明の第1実施例におけるアルゴリズムを示すフロー図

【図10】本発明の第1実施例における動作を示す波形図

【図11】本発明の第2実施例における音源発生器の時間領域における原理説明のためのタイミング図

【図12】本発明の第2実施例における音源発生器の時間領域における原理説明のためのタイミング図

【図13】本発明の第2実施例における音源発生器の周

波数領域における原理説明のためのタイミング図およびスペクトル図

【図14】本発明の第2実施例における音源発生器の周波数領域における原理説明のためのタイミング図およびスペクトル図

【図15】本発明の第2実施例における音源発生手段のブロック図

【図16】本発明の第2実施例における波形データテーブル読み出しタイミング図

【図17】本発明の第2実施例における音源発生手段のアルゴリズムを示すフロー図

【図18】本発明の第2実施例における音源発生手段のアルゴリズムを示すフロー図

【図19】本発明の第2実施例における音源発生手段のアルゴリズムを示すフロー図

【図20】本発明の第2実施例における音源発生手段のアルゴリズムを示すフロー図

【図21】本発明の第2実施例における音源発生手段の動作説明のためのタイミング図

【図22】本発明の第2実施例における音源発生手段の動作説明のためのタイミング図

【図23】本発明の第3実施例におけるエンベロープ発生手段のアルゴリズムを示すフロー図

【図24】本発明の第3実施例におけるエンベロープ波形図

【図25】本発明の第3実施例におけるエンベロープ発生手段の状態遷移図

【図26】本発明の第3実施例におけるエンベロープ発生手段のアルゴリズムを示すフロー図

【図27】本発明の第3実施例におけるエンベロープ発生手段のアルゴリズムを示すフロー図

【図28】本発明の第3実施例におけるエンベロープ発生手段のアルゴリズムを示すフロー図

【図29】本発明の第3実施例におけるエンベロープ発生手段のアルゴリズムを示すフロー図

【図30】従来のメロディ発生装置の全体構成を示すブロック図

【図31】従来のDSPによるメロディ発生装置のブロック図

【符号の説明】

- 1 メロディ発生制御器
- 2 小節カウンタ
- 3 音符カウンタ
- 4 楽譜データテーブル読み出し器
- 5 楽譜データテーブル
- 6 音符データテーブル読み出し器
- 7 音符データテーブル
- 8 波形データテーブル
- 9 波形データテーブル
- 10 波形データ選択器
- 11 音源発生器 (SG)
- 12 エンベロープパラメータテーブル
- 13 エンベロープパラメータ選択器
- 14 エンベロープ発生器 (EG)
- 15 乗算器
- 16 乗算器
- 17 音源制御器
- 18 CH1 波形データ読み出し器
- 19 CH2 波形データ読み出し器
- 20 加算器
- 21 波形データテーブル

【図2】

| 通し番号 | 音符名 | 音符No.<br>nno | 長さ<br>leng | ゲイン<br>gain | 波形<br>データNo.<br>sno | エンベ<br>ロープNo.<br>eno |
|------|-----|--------------|------------|-------------|---------------------|----------------------|
| .    | .   | .            | .          | .           | .                   | .                    |
| .    | .   | .            | .          | .           | .                   | .                    |
| .    | .   | .            | .          | .           | .                   | .                    |
| .    | .   | .            | .          | .           | .                   | .                    |
| .    | .   | .            | .          | .           | .                   | .                    |
| .    | .   | .            | .          | .           | .                   | .                    |

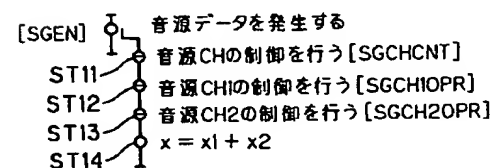
【図8】

| エンベロープ<br>No. | AR   | D1R  | D2R   | RR   | D1L  | L1M  |
|---------------|------|------|-------|------|------|------|
| 0             | 1.00 | 1.00 | 1.00  | 0.00 | 0.00 | 0.02 |
| 1             | 0.10 | 0.95 | 0.999 | 0.98 | 0.50 | 0.02 |
| 2             | 0.10 | 0.90 | 0.999 | 0.95 | 0.70 | 0.02 |

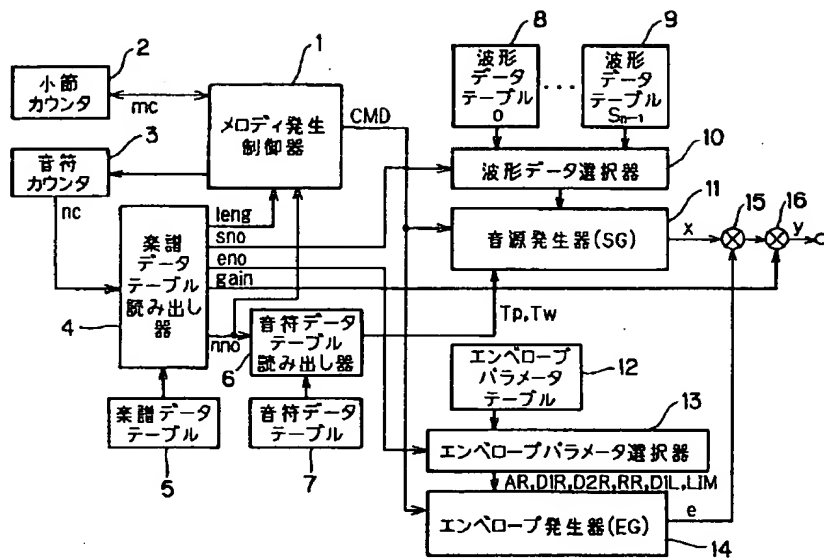
【図5】

| 音符No. | 音符名 | Tp | Tw |
|-------|-----|----|----|
| .     | .   | .  | .  |
| .     | .   | .  | .  |
| .     | .   | .  | .  |
| .     | .   | .  | .  |
| .     | .   | .  | .  |
| .     | .   | .  | .  |

【図17】



【図1】



【図3】

|    |    |    |    |     |   |   |
|----|----|----|----|-----|---|---|
| 1  | G5 | 32 | 24 | 1.0 | 0 | 0 |
| 2  | C5 | 25 | 6  | 1.0 | 0 | 0 |
| 3  | D5 | 27 | 6  | 1.0 | 0 | 0 |
| 4  | B5 | 29 | 6  | 1.0 | 0 | 0 |
| 5  | F5 | 30 | 6  | 1.0 | 0 | 0 |
| 6  | G5 | 32 | 24 | 1.0 | 0 | 0 |
| 7  | C5 | 25 | 24 | 1.0 | 0 | 0 |
| 8  | A5 | 34 | 24 | 1.2 | 0 | 0 |
| 9  | F5 | 30 | 6  | 1.2 | 0 | 0 |
| 10 | G5 | 32 | 6  | 1.2 | 0 | 0 |
| 11 | A5 | 34 | 6  | 1.2 | 0 | 0 |
| 12 | B5 | 36 | 6  | 1.2 | 0 | 0 |
| 13 | C6 | 37 | 24 | 1.2 | 0 | 0 |
| 14 | C5 | 25 | 24 | 1.2 | 0 | 0 |
| 15 | F5 | 30 | 24 | 1.0 | 2 | 0 |
| 16 | G5 | 32 | 6  | 1.0 | 2 | 0 |
| 17 | F5 | 30 | 6  | 1.0 | 2 | 0 |
| 18 | B5 | 29 | 6  | 1.0 | 2 | 0 |
| 19 | D5 | 27 | 6  | 1.0 | 2 | 0 |
| 20 | E5 | 29 | 24 | 0.8 | 2 | 0 |
| 21 | F5 | 30 | 6  | 0.8 | 2 | 0 |
| 22 | B5 | 29 | 6  | 0.8 | 2 | 0 |
| 23 | D5 | 27 | 6  | 0.8 | 2 | 0 |
| 24 | C5 | 25 | 6  | 0.8 | 2 | 0 |
| 25 | B4 | 24 | 24 | 0.8 | 2 | 0 |
| 26 | C5 | 25 | 6  | 0.8 | 2 | 0 |
| 27 | D5 | 27 | 6  | 0.8 | 2 | 0 |
| 28 | E5 | 29 | 6  | 0.8 | 2 | 0 |
| 29 | C5 | 25 | 6  | 0.8 | 2 | 0 |
| 30 | D5 | 27 | 88 | 0.7 | 2 | 0 |
| 31 | ** | 38 | 12 | 0.7 | 2 | 2 |

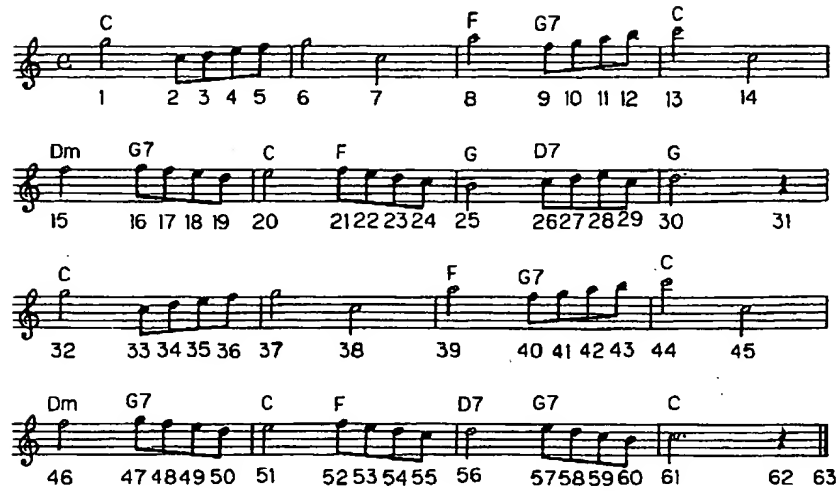
|    |    |    |    |     |   |   |
|----|----|----|----|-----|---|---|
| 32 | G5 | 32 | 24 | 1.2 | 2 | 2 |
| 33 | C5 | 25 | 6  | 1.2 | 2 | 2 |
| 34 | D5 | 27 | 6  | 1.2 | 2 | 2 |
| 35 | B5 | 29 | 6  | 1.2 | 2 | 2 |
| 36 | F5 | 30 | 6  | 1.2 | 2 | 2 |
| 37 | G5 | 32 | 24 | 1.2 | 2 | 2 |
| 38 | C5 | 25 | 24 | 1.2 | 2 | 2 |
| 39 | A5 | 34 | 24 | 1.4 | 2 | 2 |
| 40 | F5 | 30 | 6  | 1.4 | 2 | 2 |
| 41 | G5 | 32 | 6  | 1.4 | 2 | 2 |
| 42 | A5 | 34 | 6  | 1.4 | 2 | 2 |
| 43 | B5 | 36 | 6  | 1.4 | 2 | 2 |
| 44 | C6 | 37 | 24 | 1.4 | 2 | 2 |
| 45 | C5 | 25 | 24 | 1.4 | 2 | 2 |
| 46 | F5 | 30 | 24 | 1.2 | 8 | 2 |
| 47 | G5 | 32 | 6  | 1.2 | 8 | 2 |
| 48 | F5 | 30 | 6  | 1.2 | 8 | 2 |
| 49 | B5 | 29 | 6  | 1.2 | 8 | 2 |
| 50 | D5 | 27 | 6  | 1.2 | 8 | 2 |
| 51 | E5 | 29 | 24 | 1.1 | 8 | 2 |
| 52 | F5 | 30 | 6  | 1.1 | 8 | 2 |
| 53 | E5 | 29 | 6  | 1.1 | 8 | 2 |
| 54 | D5 | 27 | 6  | 1.1 | 8 | 2 |
| 55 | C5 | 25 | 6  | 1.1 | 8 | 2 |
| 56 | D5 | 27 | 24 | 1.2 | 0 | 2 |
| 57 | B5 | 29 | 6  | 1.2 | 0 | 2 |
| 58 | D5 | 27 | 6  | 1.2 | 0 | 2 |
| 59 | C5 | 25 | 6  | 1.2 | 0 | 2 |
| 60 | B4 | 24 | 6  | 1.2 | 0 | 2 |
| 61 | C5 | 25 | 38 | 1.1 | 0 | 2 |
| 62 | ** | 38 | 12 | 1.1 | 0 | 2 |
| 63 | BD | 88 | 0  | 1.0 | 0 | 2 |

【図6】

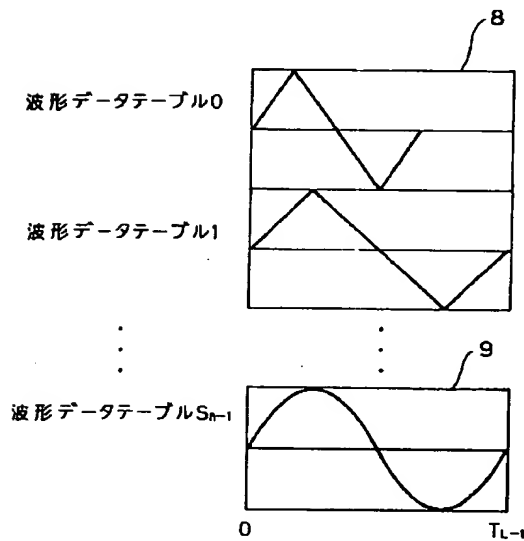
|    |     |      |     |
|----|-----|------|-----|
| 1  | C3  | 3914 | 268 |
| 2  | C#3 | 3094 | 284 |
| 3  | D3  | 3487 | 301 |
| 4  | D#3 | 3291 | 319 |
| 5  | E3  | 3107 | 338 |
| 6  | F3  | 2932 | 358 |
| 7  | F#3 | 2768 | 379 |
| 8  | G3  | 2812 | 401 |
| 9  | G#3 | 2466 | 425 |
| 10 | A3  | 2327 | 451 |
| 11 | A#3 | 2197 | 477 |
| 12 | B3  | 2073 | 506 |
| 13 | C4  | 1957 | 536 |
| 14 | C#4 | 1845 | 568 |
| 15 | D4  | 1743 | 601 |
| 16 | D#4 | 1646 | 637 |
| 17 | E4  | 1553 | 675 |
| 18 | F4  | 1466 | 715 |
| 19 | F#4 | 1384 | 758 |
| 20 | G4  | 1306 | 803 |

|    |     |      |      |
|----|-----|------|------|
| 21 | G#4 | 1233 | 851  |
| 22 | A4  | 1164 | 901  |
| 23 | A#4 | 1098 | 955  |
| 24 | B4  | 1037 | 1011 |
| 25 | C5  | 979  | 1072 |
| 26 | C#5 | 924  | 1135 |
| 27 | D5  | 872  | 1203 |
| 28 | D#5 | 823  | 1274 |
| 29 | E5  | 777  | 1350 |
| 30 | F5  | 733  | 1430 |
| 31 | F#5 | 692  | 1516 |
| 32 | G5  | 653  | 1608 |
| 33 | G#5 | 616  | 1701 |
| 34 | A5  | 582  | 1802 |
| 35 | A#5 | 549  | 1909 |
| 36 | B5  | 518  | 2023 |
| 37 | C6  | 489  | 2143 |
| 38 | **  | 0    | 0    |
| 39 | ED  | -1   | -1   |

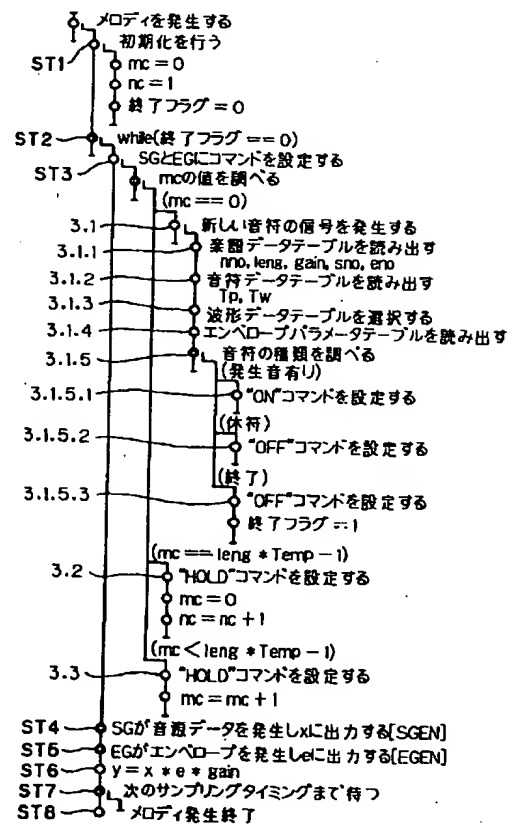
【図4】



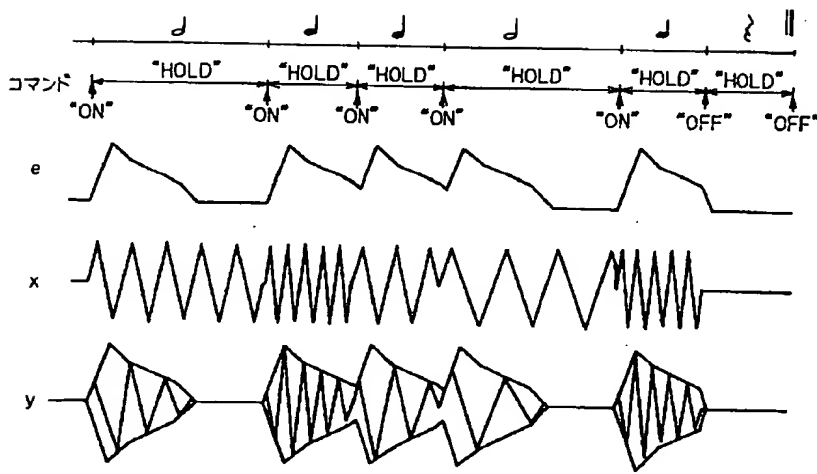
【図7】



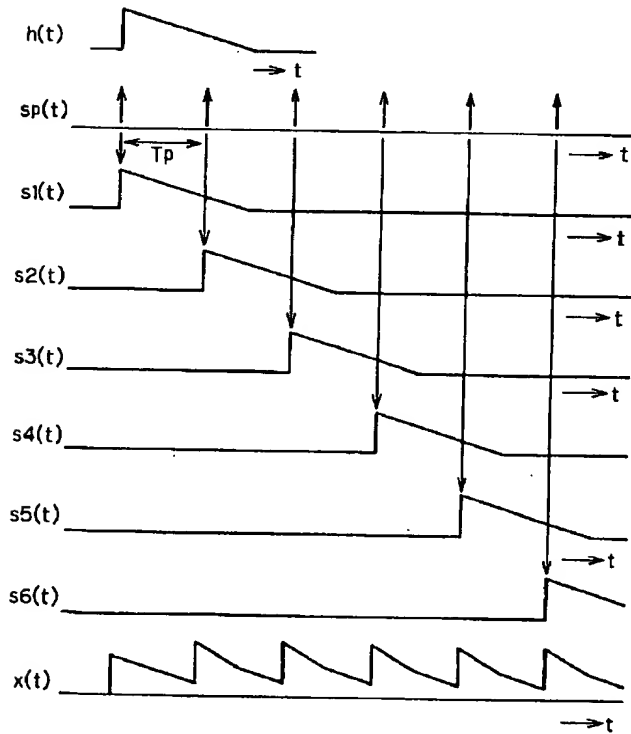
【図9】



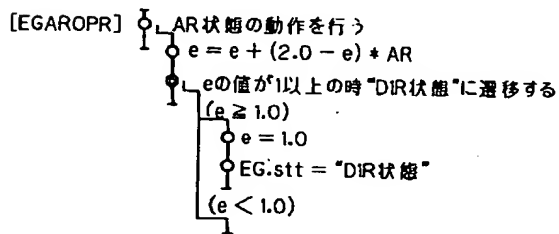
【図10】



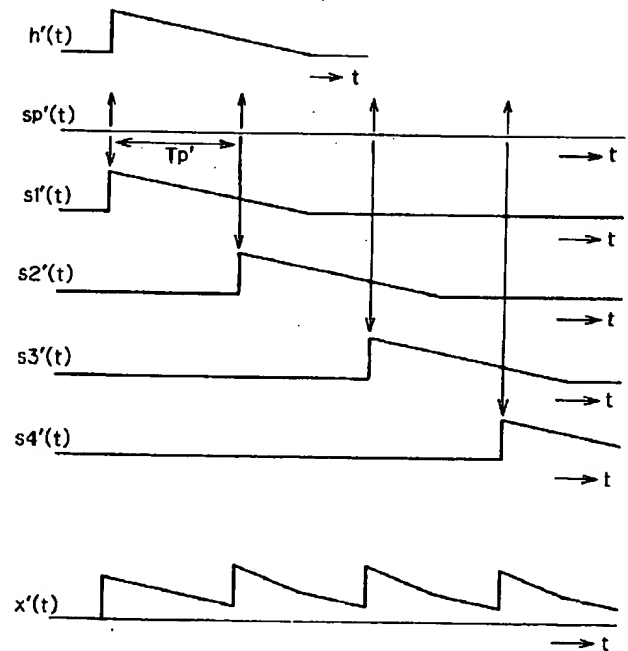
【図11】



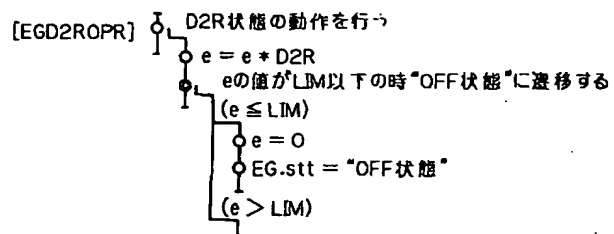
【図26】



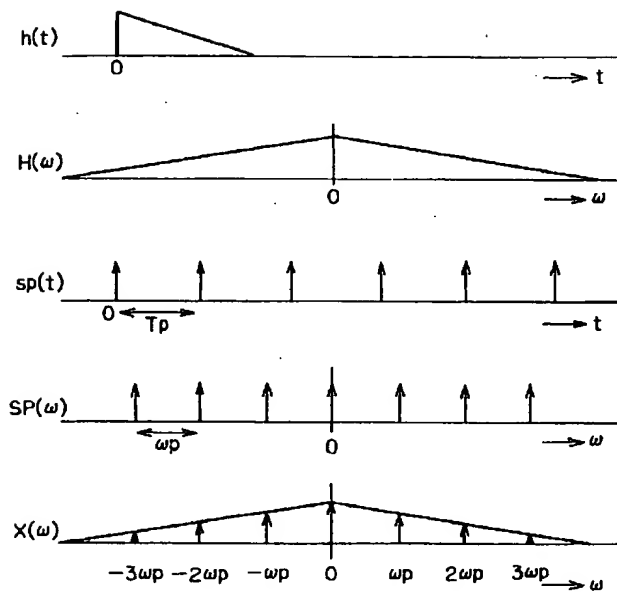
【図12】



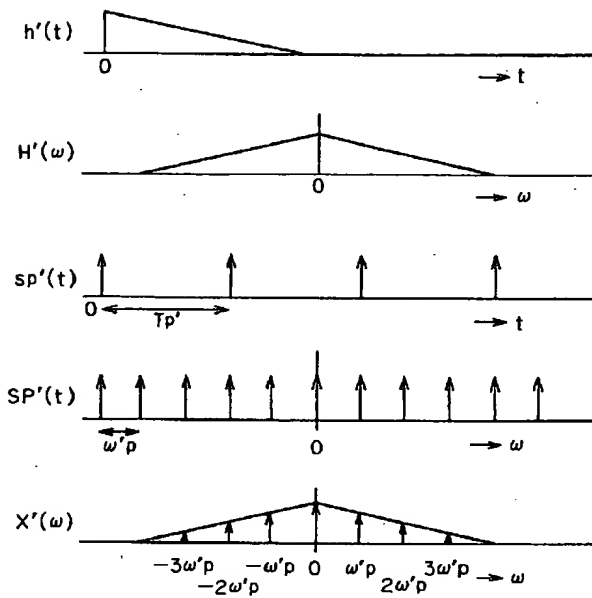
【図28】



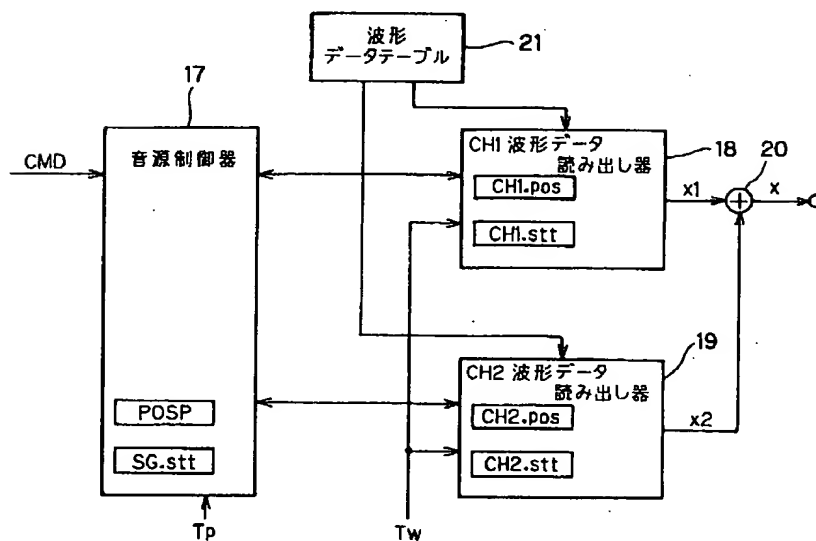
【図13】



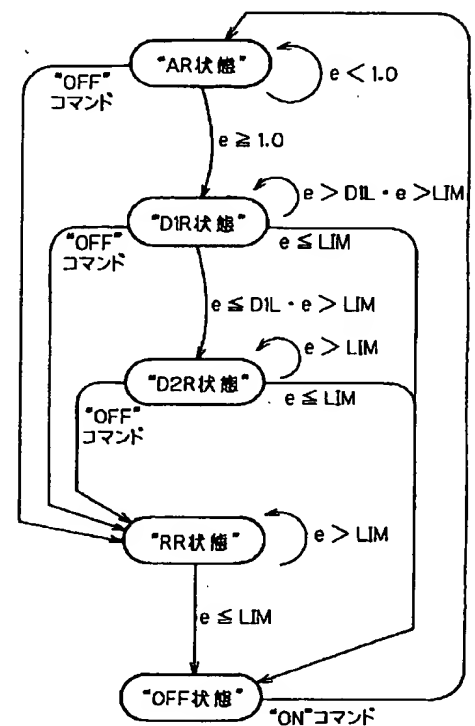
【図14】



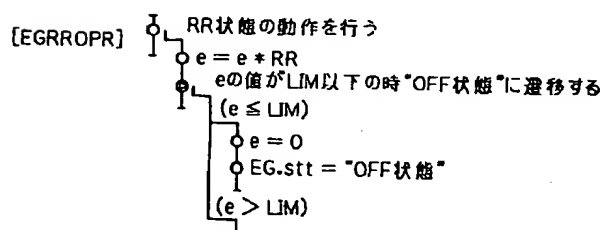
【図15】



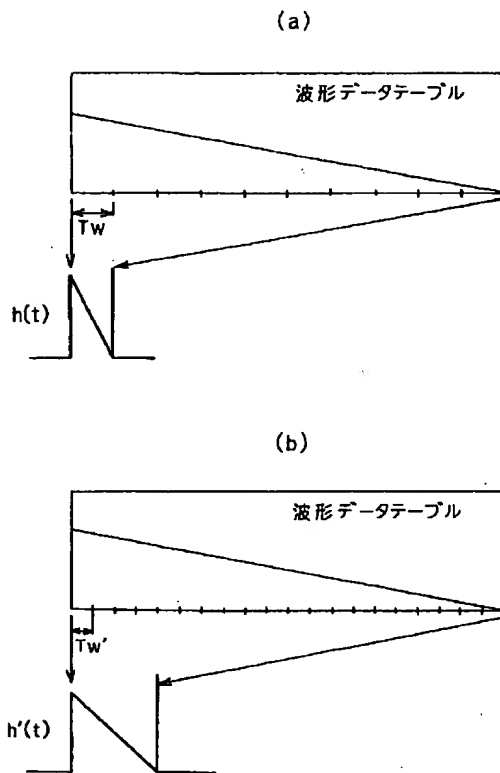
【図25】



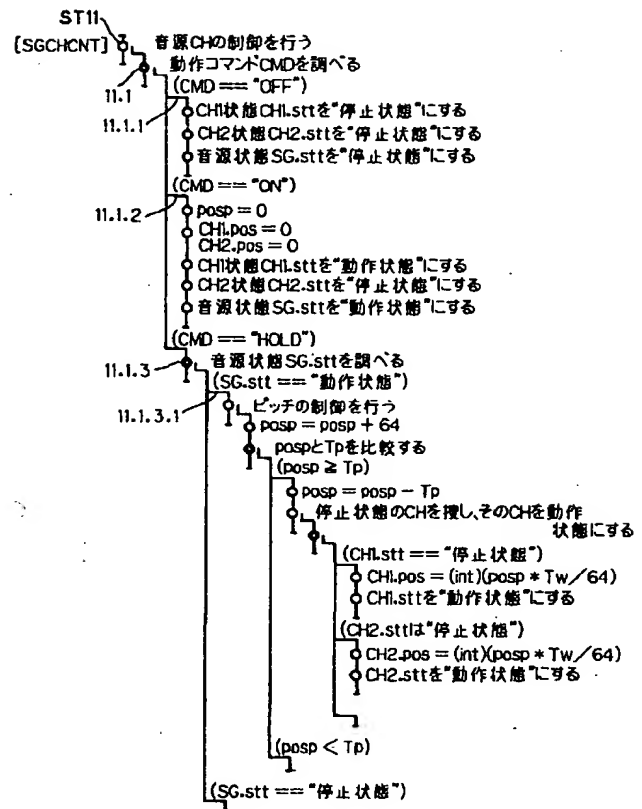
【図29】



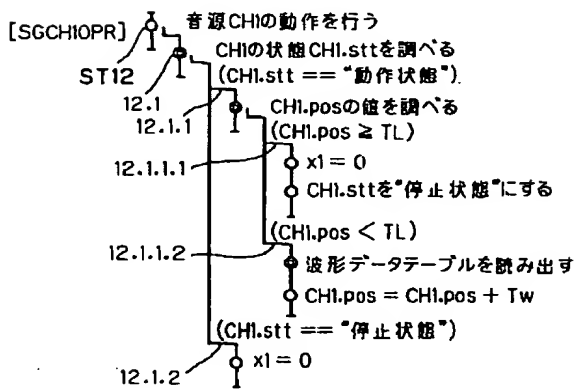
【図16】



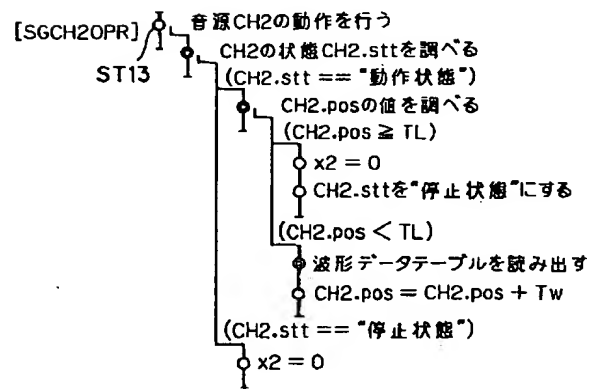
【図18】



【図19】



【図20】





サンプリング タイミング

↑ ↑ ↑ ↑ ↑ ↑

0 64 128 192 256 320

POSP 0 64 99.0 28 92 99.0 56 120

$T_p = 100$

$h(0)$   $h(10)$   $h(20)$   $h(30)$   $h(40)$

$x_1$

$x_2$

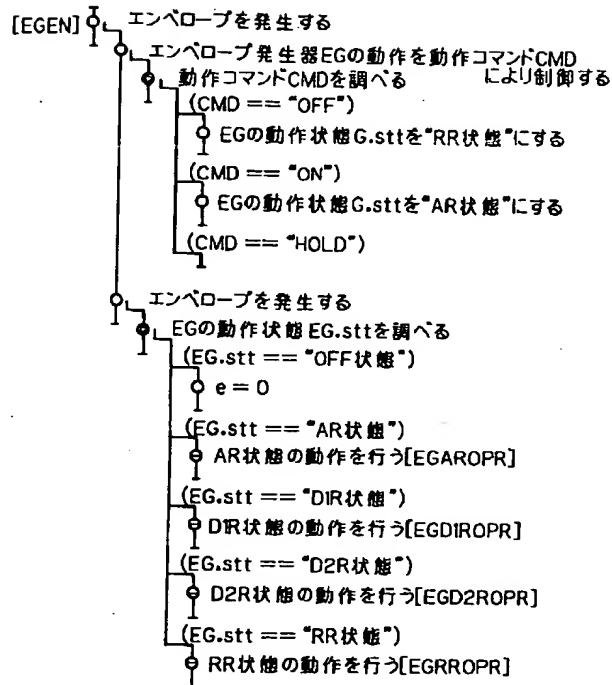
$h(4)$   $h(14)$   $h(24)$   $h(34)$

$\frac{128-100}{64} \times 10$

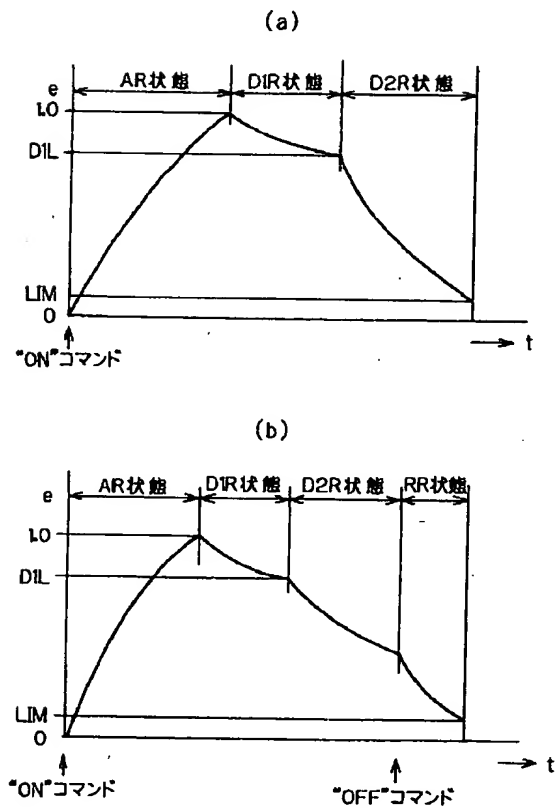
```

    graph TD
      Start([EGDIOPR]) --> SetDir((e = e * DIR))
      SetDir --> CheckLim((e ≤ LIM))
      CheckLim --> SetOff((e = 0  
EG.stt = "OFF状態"))
      SetOff --> CheckDil((e ≤ DIL))
      CheckDil --> SetDil((e = DIL  
EG.stt = "D2R状態"))
      CheckDil --> End([ ])
      CheckLim --> End
      CheckDil --> End
  
```

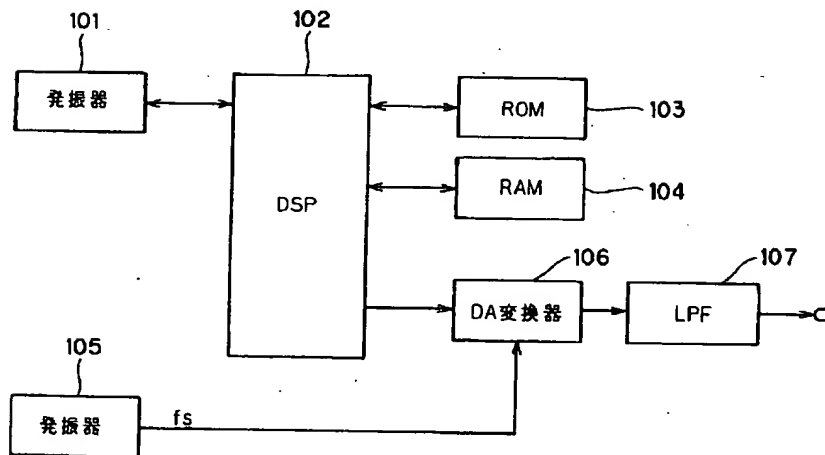
【図23】



【図24】



【図30】



【図31】

